

## Tarea 016 – Javascript – clase Bola (I)

Los resultados de todas las tareas incorporarán, además del código fuente, los comentarios precisos y necesarios para su fácil comprensión. No escatimes esfuerzos en comentar el código, es una buena práctica para aprender, además de ser muy útil para modificaciones o reutilizaciones futuras.

Sería muy buena práctica añadir el título y el enunciado del ejercicio (como comentarios) al principio del código fuente.

Añade documentos en formato Word con capturas de la salida por pantalla (al ejecutar la página) si consideras que queda más clara la resolución del ejercicio.

Deberás entregar dos versiones del resultado:

1. **Un único documento en Word sin comprimir:** que contenga todas las líneas de código, y por orden, primero el código **html**, luego el código **css** y por último el código **js**. Este archivo servirá para que el profesor haga anotaciones y/o correcciones a los alumnos.
2. **Los archivos de código fuente:** en un único archivo, ya sea de extensión: **html**, **js**, o **css**. Si precisas entregar varios archivos, comprímelos en un único **zip**. (No admito rar).

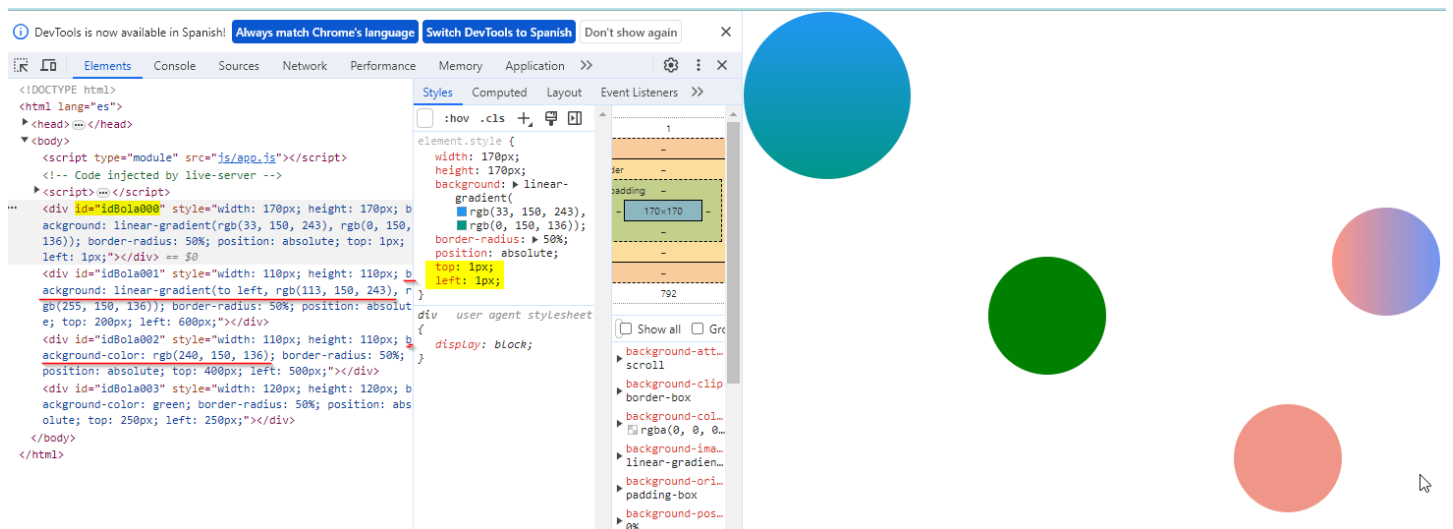
El nombre del archivo entregado comenzará por tu nombre seguido por TareaXXX. Ejemplo: **federicoTarea014.zip**

### 016 – clase Bola (I)

Enunciado:

Se pide

1. Elaborar una clase de nombre Bola para generar un número indeterminado de objetos de esta clase.
2. Utilizar la clase anterior para mostrar en el navegador unas cuantas bolas.



En la imagen de arriba se ven 4 objetos creados utilizando el código siguiente:

```
import { misFunciones } from "../libreria/misfunciones.js";
import { Bola } from "../clases/clase_bola.js";

//la línea de abajo es un ejemplo de uso de una función de la librería que debes
// realizar para que cada objeto de la clase Bola tenga un id="idBolaXXX" siendo XXX
// el número (con tres cifras) de índice en el array de cada objeto
console.log(misFunciones.formatearA3cifras(5)); // muestra "005" en consola

//constructor(radius, posX, posY, colorBola, linearGradient)

new Bola(85,1,1,'red', 'linear-gradient(to bottom, #2196f3, #009688)');
new Bola(55,200,600,'red', 'linear-gradient(to left, #7196f3, #ff9688)');
```

```
new Bola(55,400,500,'#f09688');  
new Bola(60, 250,250,"green")  
Bola.arrayBolas.forEach(bola=>bola.visualizar());  
  
console.log(Bola.arrayBolas);
```

Fíjate que, al instanciar los objetos, en el constructor puedes poner 4 o 5 argumentos. Si pones el 5º argumento, éste será el valor de la propiedad “background”, que dejará sin efecto la propiedad “background-color”. (Mira en la imagen lo subrayado en rojo).

Utiliza las propiedades CSS que se observan en las DevTools de la imagen anterior.

Consideraciones:

- Las bolas son divs rectangulares con border-radius al 50%
- El tamaño del div debes calcularlo teniendo en cuenta que el primer argumento del constructor es el radio.
- La posición del div en el documento, se consigue con la propiedad CSS **position: absolute**
- Las propiedades **top** y **left** se refieren a la esquina superior izquierda del div. Mira en la imagen, estos valores llevan “px” detrás del número.
- En un documento del navegador, la coordenadas (1,1) se refieren a la esquina superior izquierda.